Compressing the Input for CNNs with the First-Order Scattering Transform

Edouard Oyallon, Eugene Belilovsky, Sergey Zagoruyko, Michal Valko

Compressing the Input for CNNs with the First-Order Scattering Transform

Edouard Oyallon,[1,4,5] Eugene Belilovsky,[2] Sergey Zagoruyko,[3] Michal Valko[4]

[1]CentraleSupelec, Université Paris-Saclay
[2]MILA, University of Montreal
[3]WILLOW – Inria Paris, [4]SequeL – Inria Lille, [5]GALEN – Inria Saclay

Abstract. We study the *first-order* scattering transform as a candidate for reducing the signal processed by a *convolutional neural network* (CNN). We study this transformation and show theoretical and empirical evidence that in the case of natural images and sufficiently small translation invariance, this transform preserves most of the signal information needed for classification while *substantially reducing* the spatial resolution and total signal size. We show that cascading a CNN with this representation performs on par with ImageNet classification models commonly used in downstream tasks such as the ResNet-50. We subsequently apply our trained hybrid ImageNet model as a base model on a detection system, which has typically larger image inputs. On Pascal VOC and COCO detection tasks we deliver *substantial improvements in the inference speed* and training memory consumption compared to models trained directly on the input image.

Keywords: CNN, SIFT, image descriptors, first-order scattering

1 Introduction

Convolutional neural networks (CNNs) for supervised vision tasks learn often from raw images [1] that could be arbitrarily large. Effective reduction of the spatial dimension and total signal size for CNN processing is difficult. One way is to learn this dimensionality reduction during the training of a supervised CNN. Indeed, the very first layers of standard CNNs play often this role and reduce the spatial resolution of an image via *pooling* or *stride operators*. Yet, they generally maintain the input layer sizes and even increase it by expanding the number of channels. These pooling functions can correspond to a linear pooling such as wavelet pooling [2], a spectral pooling [3], an average pooling, or a non-linear pooling such as ℓ^2-pooling [4], or max-pooling. For example, the two first layers of an AlexNet [5], a VGG [6] or a ResNet [7] reduce the resolution respectively by 2^3, 2^1, and 2^2, while the dimensionality of the layer is increased by a factor 1.2, 5.3, and 1.3 respectively. This spatial size reduction is important for computational reasons because the complexity of convolutions is quadratic in spatial size while being linear in the number of channels. This suggests that reducing the input size to subsequent CNN layers calls for a careful design. In

this work, we (a) analyze a generic method that, *without learning*, reduces *input size* as well as *resolution* and (b) show that it *retains enough information* and structure that permits applying a CNN to obtain competitive performance on classification and detection.

Natural images have a lot of redundancy, that can be exploited by finding a frame to obtain a sparse representation [8,9]. For example, a wavelet transform of piece-wise smooth signals (e.g., natural images) leads to a multi-scale and sparse representation [10]. This fact can be used for a compression algorithm [11]. Since in this case the most of the information corresponds to just a few wavelet coefficients, a transform coding can be applied to select them and finally quantize the signal, which is consequently a more compact representation. Yet, this leads to variable signal size and thus this method is not amenable for CNNs that require a constant-size input. Another approach is to select a subset of these coefficients, which would be a linear projection. Yet, a linear projection would imply an unavoidable loss of significant discriminative information which is not desirable for vision applications. Thus, we propose to use a *non-linear* operator to reduce the signal size and we justify such construction.

Prior work has proposed to input predefined features into CNNs or neural networks. For example, [12] proposed to apply a deep neural network on Fisher vectors. This approach relies on the extraction of overlapping descriptors, such as SIFT, at irregular spatial locations and thus does not permit a fixed size output. Moreover, the features used in these models increase the signal size. In [13], wavelets representations are combined at different layer stages, similarly to DenseNet [14]. [15] proposes to apply a 2D Haar transform that leads to subsampled representation by a factor of 2^1 but is limited to this resolution. Concurrent to our work, [16] proposed to train CNNs on top of raw DCT to improve inference speed by reducing the spatial resolution, yet this transformation is orthogonal and thus preserves the input size. Moreover, [17] proposes to input *second-order* scattering coefficients to a CNN, that are named *hybrid scattering networks*, which lead to a competitive performance on datasets such as ImageNet. The scattering transform is a *non-linear* operator based on a cascade of wavelet transforms and modulus non-linearity which are spatially averaged. This leads to a reduction in the spatial resolution of the signal. However, although the second-order scattering representation is more discriminative, it produces a larger signal than the original input size.

In this work, we also input predefined features into CNNs, *but* with the explicit goal of an initial stage producing a compressed representation that is still amenable to processing by a CNN. In particular, we show that the first-order scattering representation is a natural candidate for several vision tasks. This descriptor leads to high accuracy on large-scale classification and detection while it can be computed much faster than its second-order counterpart because it requires fewer convolutions. As explained in [18], this descriptor is similar to SIFT and DAISY descriptors that have been used as feature extractors in many classical image classification and detection systems [19,20]. In this paper, we show that in the case of hybrid networks [17,12], using the *first-order scattering*

only can have favorable properties with respect to the second-order ones and possibly higher-order ones.

The core of our paper it the analysis and justification of the combination of first-order scattering and CNNs. We support it both with theoretical and numerical arguments. In Section 2, we justify that first-order scattering with small-scale invariance reduces the spatial resolution and signal while preserving important attributes. First, we motivate the first-order scattering from a dimensionality reduction view in Section 2.1. Then, in Section 2.2, we illustrate the negligible loss of information via a good reconstruction of synthetic signals and natural images using only a first-order scattering. Next, in Section 3 we present our experiments[1] on challenging datasets. We demonstrate competitive performance with ImageNet models commonly used in transfer learning in Section 3.1. In Section 3.2 we show on COCO and Pascal VOC detection tasks that these base networks can lead to improvements in terms of inference speed and memory consumption versus accuracy.

2 First-order scattering

In this section, we motivate the construction of a first-order *scattering transform* from a compression perspective. Indeed, a scattering transform is traditionally built as a representation that preserves high-frequency information, while building stable invariants w.r.t. translations and deformations. While using the same tools, we adopt a rather different take. We show theoretically and numerically that a first-order *scattering transform* builds limited invariance to translation, reduces the size of an input signal, preserves most of the information needed to discriminate and reconstruct a natural image. Note also that this representation is able to discriminate *spatial* and *frequency variations* of natural images. In this section, we deal with *Gábor wavelets* [21] since their analysis is simpler, while for the experiments we will use modified Gábor wavelets, namely *Morlet wavelets* [17] for the sake of comparison. We show that the first-order scattering transform does not lose significant signal characteristics of natural images, by providing reconstruction examples obtained via a mean-square error minimization. In particular, we demonstrate this property on *Gaussian blobs* as a simplified proxy for natural images.

2.1 A reduction of the spatial resolution

Definition A scattering first-order transform [22] is defined from a *mother wavelet* ψ and a *low-pass filter* ϕ. An input signal x is filtered by a collection of dilated band-pass wavelets obtained from ψ, followed by a *modulus* and finally averaged by a *dilation* of ϕ. The wavelets we chose decompose the signal in a basis in which transient structure of a signal is represented more compactly. We describe the construction of each filter and justify the necessity of each operator.

[1] code available at https://github.com/edouardoyallon/pyscatlight

First, let us fix an integer J that specifies the window length of the low-pass filter. For the sake of simplicity, we consider Gábor filters [21]. These filters provide a good localization tradeoff between *frequency* and *space planes*, due to Heisenberg uncertainty principle [9]. Thus, having

$$\kappa(\omega) \triangleq e^{-2\sigma_0^2\|\omega\|^2}$$

for a fixed bandwidth σ_0 and a slant s that discriminates angles, we set for $\omega = (\omega_1, \omega_2)$,

$$\widehat{\psi}(\omega) \triangleq \kappa\left(\left(\omega_1, \frac{\omega_2}{s}\right) - \omega_0\right) \quad \text{and} \quad \widehat{\phi}(\omega) \triangleq \kappa(\omega).$$

The frequency plane (and in particular the image frequency circle of radius π) needs to be covered by the support of the filters to avoid an information loss. This issue is solved by the action of the Euclidean group on ψ via rotation $r_{-\theta}$ and dilation by $j \leq J$,

$$\psi_{j,\theta}(u) = \frac{1}{2^{2j}}\psi\left(r_{-\theta}\frac{u}{2^j}\right) \quad \text{and} \quad \phi_J(u) = \frac{1}{2^{2J}}\phi\left(\frac{u}{2^J}\right).$$

In this case, each wavelet $\psi_{j,\theta}$ has a bandwidth of $1/(2^j\sigma_0)$ and its central frequency is $2^j r_{-\theta}\omega_0$. If a filter has a compact support in the frequency domain, then due to Nyquist principle, we can reduce the spatial sampling of the resulting convolution. We do this approximation in the case of Gábor filters. As we shall see, this localization in frequency is also fundamental because it permits to obtain a *smooth envelope*. The parameters $j \leq J$ and $\theta \in \Theta$ are discretized and σ_0 is adjusted such that a wavelet transform preserves all the energy of $\widehat{x}$, characterized by

$$\exists \varepsilon_0 \geq 0, \forall \omega, \|\omega\| < \pi : \; 1 - \varepsilon_0 \leq \sum_{j \leq J, \theta \in \Theta} \left|\widehat{\psi}_{j,\theta}(\omega)\right|^2 + \left|\widehat{\phi}_J(\omega)\right|^2 \leq 1 + \varepsilon_0.$$

As a result, the transform is bi-Lipschitz and the magnitude of ε_0 determines the conditioning of the wavelet transform. An ideal setting is $\varepsilon_0 = 0$, for which the transform is an isometry which gives a one-to-one mapping of the signal while preserving its ℓ^2-norm. Applying a convolution with these wavelets followed by a modulus removes the phase of a signal and thus should lead to a loss of information. In fact, [23] proves that it is possible to reconstruct a signal from the modulus of its wavelet transform up to a global translation with *Cauchy wavelets*. Furthermore, there exists an algorithm of reconstruction [24], with stability guarantees and extension to other class of wavelets. Consequently, the modulus of a wavelet transform does not lead to a significant loss if applied appropriately. Additionally, [22] demonstrates that this representation is stable to deformations, which permits building invariants to deformations, convenient in many vision applications. We now explain how the dimensionality reduction occurs.

The scattering first-order transform S [22] parametrized by J is[2] defined as

$$Sx(u) = \left\{ |x \star \psi_{j,\theta}| \star \phi_J(2^J u), x \star \phi_J(2^J u) \right\}_{\theta \in \Theta, j \leq J}.$$

The low-pass filter ϕ_J builds a transformation that is locally invariant to translation up to 2^J. Therefore, it reduces the spatial sampling of the signal by a factor of 2^J. This also means that when discretized image of length N represented by N^2 coefficients, is filtered by the low-pass filter ϕ_J, the signal is represented by $N^2/2^{2J}$ coefficients. Consequently, the number of coefficients used to represent Sx is

$$(1 + |\Theta| J) \frac{N^2}{2^{2J}}.$$

In our case, we use $|\Theta| = 8$, because it permits obtaining a good covering of the frequency plane, and thus, the input signal x is compressed via Sx if $J \geq 3$. The low-pass filtering implies a necessary loss of information because it discards some high-frequency structure and only retains low frequencies which are more invariant to translation. It is fundamental to evaluate the quality of this compressed representation in order to validate that enough information is available for supervised classifier such as CNNs, which is what we do next.

Preserving signal information via modulus We evaluate the loss of information due to the low-pass filtering, which captures signal attributes located in the low-frequency domain. Notice that there would be no loss of information if the Fourier transform of the wavelet-modulus representation was located in a compact domain included in the bandwidth of ϕ_J. Unfortunately, this property is not guaranteed in practice.

Nonetheless, Gábor wavelets are *approximately analytic* which implies that when convolved with a signal x, the resulting envelope is smoother [25,9,26,22,27]. A smooth envelope of the signal implies that a significant part of its energy can be captured and preserved by a low-pass filter [22]. Furthermore, under limited assumptions of point-wise regularity on x, if the signal does not vanish, it is possible to quantify this smoothness, as done in [27]. Informally, for a translation $x_a(u) \triangleq x(u - a)$ by a of x, it means that if $\|a\| \ll 1$, then we imply that

$$|x_a \star \psi|(u) \approx |x \star \psi|(u).$$

Here, we simply give some explicit constant w.r.t. the stability to translation, that we relate to the envelope of ψ. Indeed, the Gábor filter ψ concentrates its energy around a central frequency ω_0,

$$\exists \eta_0 > 0, \exists \omega_0, \varepsilon \geq 0, \forall \omega, \quad \|\omega - \omega_0\|_2 > \eta_0 \implies |\widehat{\psi}(\omega)| \leq \varepsilon.$$

First-order scattering incorporates more information if the modulus operator has smoothed the signal. To this end, we characterize the stability w.r.t. translations in the case of Gábor wavelets. In particular, we provide the following Lipschitz bound w.r.t. translations.

[2] in the following, we omit the dependence w.r.t. the scale J

Proposition 1. *For any signal $x \in \ell^2$,*

$$\|x_a \star \psi - e^{-i\omega_0^\intercal a} x \star \psi\| \leq \|x\| \left(\|\eta_0\|\|a\| + \widetilde{\varepsilon}(\|a\|)\right),$$

where $\widetilde{\varepsilon}$ is a term of the order of ε.

Proof. Observe that

$$\|x_a \star \psi - e^{-i\omega_0^\intercal a} x \star \psi\|^2 = \int \left| \left(e^{-i\omega^\intercal a} - e^{-i\omega_0^\intercal a}\right) \widehat{\psi}(\omega)\widehat{x}(\omega) \right|^2 \mathrm{d}\omega \text{ via Parseval identity}$$

$$\leq 4\varepsilon^2 \|x\|^2 + \int_{\|\omega-\omega_0\|<\eta_0} \left|(e^{-i\omega^\intercal a} - e^{-i\omega_0^\intercal a})\widehat{\psi}(\omega)\right|^2 \left|\widehat{x}(\omega)\right|^2 \mathrm{d}\omega.$$

(note that $x \mapsto e^{ix}$ is 1-Lipschitz, thus we apply the Cauchy-Schwartz inequality)

$$\leq \|x\|^2 \left(4\varepsilon^2 + \|a\|^2\eta_0^2\right).$$

Taking the square root finishes the proof.

We note that this inequality is near-optimal, for Gábor wavelets, if $x(u) = \delta_0$ is a Dirac in 0, then $|e^{i\omega_0^\intercal a}\psi(a) - \psi(0)| \sim \|x\|\|a\|\eta_0$. Observe that dilating the mother wavelet ψ to ψ_j is equivalent to dilating the bandwidth η_0 to $2^{-j}\eta_0$. Following the reasoning, low-frequency Gábor wavelets are more likely to be invariant to a translation.

Proposition 1 characterizes the Lipschitz stability w.r.t. translations and indicates that the more localized a Gábor wavelet is, the more translation-stable is the resulting signal. This way, we justify Gábor wavelets as a great candidate for a wavelet transform with a smooth modulus with limited assumptions on x.

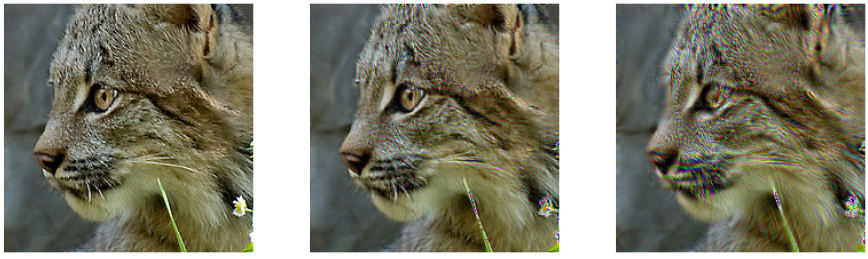
Note that using only small-bandwidth Gábor wavelets instead of dilated ones should be avoided because it would lead to significantly more filters. Furthermore, [22] shows that those filters will be more unstable to deformations, such as dilation, which is not desirable for vision applications.

Despite the stability to translation, there is no guarantee that the first-order scattering preserves the complete energy of the signal. The next section characterizes this energy loss via an image model based on Gaussian blobs and a reconstruction algorithm for natural images.

2.2 Information loss

We now characterize the information loss for natural images in two ways. First, we perform an empirical reconstruction of an image from its first-order scattering coefficients, as done for the second order in [28,29] and observe that for natural images, we can indeed obtain an effective reconstruction of the first-order scattering. This is a strong indication that the relevant signal information is preserved. Second, we consider a generic signal model and show that for relatively low-scale factors J, the reconstruction is practically achieved.

(a) **middle**: PSNR $\approx 26dB$, right: PSNR $\approx 20dB$



(b) **middle**: PSNR $\approx 23dB$, right: PSNR $\approx 19dB$

Fig. 1: Reconstructed images from first-order scattering coefficients, $J = 3, 4$ and their PSNR. Color channels can be slightly translated leading to artifacts. (**left**) original image x (**middle**) reconstruction $\widetilde{x}_3$ from Sx, $J = 3$ (**right**) reconstruction $\widetilde{x}_4$ from Sx, $J = 4$. This demonstrates that even complex images can be reconstructed for $J = 3$, while dividing the spatial resolution by 2^3.

Reconstruction Following [28,29], we propose to reconstruct an input image x from its first-order scattering Sx coefficient of scales J, via a ℓ^2-norm minimization

$$\widetilde{x}_J = \inf_y \|Sx - Sy\|. \tag{1}$$

We use a gradient descent as all the operators are weakly differentiable and we analyze the reconstructed signal. Figure 1 compares the reconstruction of a natural image with the first-order scattering for the scales $J = 3$ and $J = 4$. In our experiments, we optimize for this reconstruction with ADAM with an initial learning rate of 10 during 10^3 iterations, reducing by 10 the learning rate every 2×10^2 iterations. We measure the reconstruction error of $\widetilde{x}_J$ from an original image x in terms of relative error, defined as

$$\mathrm{err}_J(x) = \frac{\|S\widetilde{x}_J - Sx\|}{\|Sx\|}.$$

In other words, we evaluate how close the scattering representation of an image is to its reconstruction. We stop the optimization procedure as soon as we

get $\mathrm{err}_J(x) \sim 2 \times 10^{-3}$. In the case $J = 3$, observe that the important and high-frequency structure of the signals, as well as their spatial localization, are preserved. On the contrary, when $J = 4$, the fine-scale structure is neither well reconstructed nor correctly located, which tends to indicate that $J \geq 4$ might not be a good-scale candidate for S. We now characterize this loss more precisely on a model based on blobs.

Gaussian blob model Explicit computation of scattering coefficients for general signals is difficult because a modulus is a non-linear operator that usually leads to non-analytic expressions. Therefore, we consider a simplified class of signals [30] for which computations are exact and analytical. For a symmetric matrix $\mathbf{\Sigma}$, we consider the unnormalized signal

$$\widehat{x}_{\mathbf{\Sigma}}(\omega) \triangleq e^{-\omega^\intercal \mathbf{\Sigma} \omega}.$$

Figure 2 shows several signals belonging to this class. Such signals correspond to blobs or lines as on Figure 2, which are frequent in natural images [31]. We apply our reconstruction algorithm and we explain why the reconstructions is challenging.

In particular, we prove the following proposition that is derived from convolutions between Gaussians and permits to compute their first-order scattering coefficients. Intuitively, this proposition says that for a particular class of signals, we can get their exact reconstruction from their first-order scattering coefficients. Note that for large values of J, the reconstruction is numerically infeasible.

Proposition 2. *For any symmetric* $\mathbf{\Sigma}$*,* j*, and* θ*,*

$$|x_{\mathbf{\Sigma}} \star \psi_{j,\theta}|(u) \propto \left(x_{\mathbf{\Sigma}} \star |\psi_{j,\theta}|\right)(u).$$

Proof. Without loss of generality, we prove the result for $\widehat{\psi}(\omega) \triangleq e^{-\|\mathbf{\Gamma}\omega - b\|^2}$, where $\mathbf{\Gamma}$ is invertible and $b \in \mathbb{R}^2$. Then, $\widehat{|\psi|}(\omega) \propto e^{-\|\mathbf{\Gamma}\omega\|^2}$. Let $\Delta(u) \triangleq x_{\mathbf{\Sigma}} \star \psi(u)$. Then by definition,

$$\widehat{\Delta}(\omega) \propto e^{-\omega^\intercal (\mathbf{\Sigma} + \mathbf{\Gamma}^\intercal \mathbf{\Gamma})\omega + 2\omega^\intercal \mathbf{\Gamma}^\intercal \mathbf{\Gamma} b}.$$

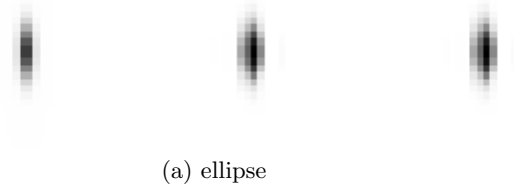
As $\mathbf{\Gamma}^\intercal \mathbf{\Gamma} \succ 0$, we can set $\widetilde{b} \triangleq \left(\mathbf{\Sigma} + \mathbf{\Gamma}^\intercal \mathbf{\Gamma}\right)^{-1} \mathbf{\Gamma} b$. Then, the result comes from an inverse Fourier transform applied to

$$\widehat{\Delta}(\omega) \propto e^{-(\omega - \widetilde{b})^\intercal (\mathbf{\Sigma} + \mathbf{\Gamma}^\intercal \mathbf{\Gamma})(\omega - \widetilde{b})}.$$

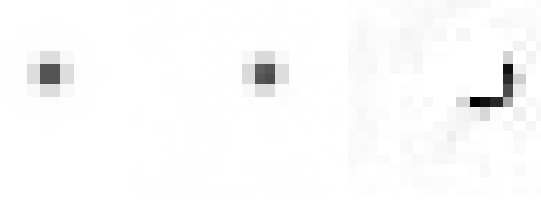
Therefore, the first-order scattering coefficients are given by

$$S x_{\mathbf{\Sigma}} \propto \{x_{\mathbf{\Sigma}} \star (|\psi_{j,\theta}| \star \phi_J), x_{\mathbf{\Sigma}} \star \phi_J\}_{\theta \in \Theta, j \leq J}.$$

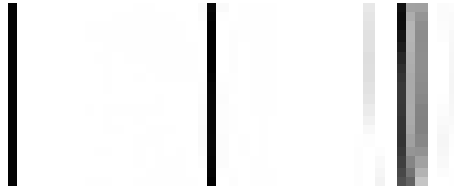
A naïve inversion of the first-order scattering coefficients would be an inversion of the convolution with $|\psi_{j,\theta}| \star \phi_J$ which is unfortunately poorly conditioned for large values of J since this filter is a low-pass one. However, solving the



(a) ellipse



(b) small blob



(c) line

Fig. 2: Reconstruction of different signals of type $x_{\mathbf{\Sigma}}$. (**left**) original image (**middle**) reconstruction via for $J = 3$ (**right**) reconstruction for $J = 4$

optimization (1) leads to a different solution due to the presence of the modulus during the gradient computation. For $J \leq 3$, it is possible to recover the original signal as shown in Figure 2. Nevertheless, there is a lack of spatial localization for $J \geq 4$ due to the averaging ϕ_J, that we observe during our reconstruction experiment. This confirms our choice of $J = 3$ for the remainder of the paper.

3 Numerical experiments

We perform numerical experiments using first-order scattering output as the input to a CNN. Our experiments aim to both validate that the first-order scattering can preserve the key signal information and highlight the practical importance of it. In particular, we find that we obtain performance close to some of the state-of-the-art systems *while improving inference speed* and memory consumption during training by light years..

3.1 ImageNet classification experiments

We first describe our image classification experiments on the challenging ImageNet dataset. Each of our experiments is performed using standard hyperparameters *without a specific adaptation* to our hybrid architecture. Extensive architecture search for the first-order scattering input is not performed and we believe that these results can be improved with resources matching the architectures and hyperparameters developed for natural images.

ImageNet ILSVRC2012 is a challenging dataset for classification. It consists of 1k classes, 1.2M large colored images for training and 400k images for testing. We demonstrate that our representation does not lose significant information for classification by obtaining a competitive performance on ImageNet. We follow a standard procedure training procedures [32,17,7]. Specifically, we applied standard data augmentation and crop input images to a size of 224^2. The first order scattering then further reduces this to a size of 28×28. We trained our CNNs by stochastic gradient descent (SGD) with the momentum of 0.9, weight decay of 10^{-4} and batch size of 256 images, trained for 90 epochs. We reduce the learning rate by 0.1 every 30 epochs. At test time, we rescale the images to 256^2 and crop an image of size 224^2.

To construct our scattering hybrid networks, we stay close to an original CNN reference model. In particular, we build our models out of the ResNets [7] and WideResNets [32] models. A typical ResNet consists of an initial layer followed by $K = 4$ so-called layer groups that in turn consist of $[n_1, \dots, n_K]$ residual blocks, where n_i specifies the number of blocks in each layer group. Furthermore, the width in each blocks is a constant and equal to $[w_1, \dots, w_K]$. Similarly to [17], an initial convolutional layer is applied to increase the number of channels from $3 \times (1 + 8J) = 75$ to w_1. A stride of 2 is applied at the initial layer of the blocks $k \geq 2$, to reduce the spatial resolution. Each of the residual blocks contains two convolutional operators, except when a stride of 2 is applied in order to replace the identity mapping, in which case there are three convolutional operators, as done in [7]. In the following, we refer to ScatResNet-L as the architecture with L convolutional operators. As discussed we used $J = 3$, as done in [17].

In our first experiment, we aim to directly compare to the results of [17] which use the second-order scattering. Thus we use the same structure that applies $K = 2$ layer groups on the scattering input instead of the typical 4. This architecture was called the ScatResNet-10 [17], and has $[2, 2]$ layers of width $[256, 512]$. The number of parameters is about 12M in both cases. Notice that the number of parameters varies only since the initial number of input channels change. Table 1 reports similar accuracy for order 1 and order 2 scattering, which indicates that if enough data is available and there is a small invariance to translation J, then for natural image classification, the order 2 does not provide significantly more information that can be exploited by a CNN.

Now we demonstrate that the scattering first-order transform continues to scale further when applying more sophisticated networks. Note that this would not have been possible with a second-order scattering in a reasonable time. In our case, we avoid computing many convolutions. Scaling to these modern networks

permits us to apply the scattering in the subsequent section to common computer vision tasks that require a base network from ImageNet, and where the smaller input size leads to gains in speed and memory.

The models we construct are the ScatResNet-50, based on the ResNet50 architecture, and the WideScatResNet-50-2 based on the wide ResNet that expands the channel width and leads to competitive performance [32]. Since the scattering input starts at a much lower resolution, we bypass the first group of the typical ResNet, which normally consists of $K = 4$ layer groups and reduce the number of groups to $K = 3$. A typical ResNet50 has 16 residual blocks distributed among the 4 layer groups. We maintain the same number of total residual blocks and thereby layers as in the ResNet50, redistributing them among the three groups using $[5, 8, 3]$ blocks. As in their non-scattering analogue we apply bottleneck blocks [7]. The width of the blocks for ScatResNet-50 and WideScatResNet-50-2 are $[128, 256, 512]$ and $[256, 512, 1024]$, which matches the widths of groups 2 through 4 of their non-scattering counterparts.

Table 1: Accuracy on ImageNet. Note that scattering based models have input sizes of $28\times28\times75$ while the normal ImageNet models are trained on $224\times224\times3$.

Architecture	Top 1	Top 5	#params
Order 1,2 + ScatResNet-10 [17]	68.7	88.6	12.8M
Order 1 + ScatResNet-10	67.7	87.7	11.4M
Order 1 + ScatResNet-50	74.5	92.0	27.8M
Order 1 + WideScatResNet-50-2	76.2	92.8	107.2M
ResNet-50 (`pytorch`)	76.1	92.9	25.6M
ResNet-101(`pytorch`)	77.4	93.6	45.4M
VGG-16 [6]	68.5	88.7	138M
ResNet-50 [7]	75.3	92.2	25.6M
ResNet-101	76.4	92.9	45.4M
WideResNet50-2 [32]	77.9	94.0	68.9M
ResNet-152	77.0	93.3	60.2M

Table 1 indicates that the performance obtained by those architectures can be competitive with their respective reference architectures for classification. We compare to the reference models trained using the same procedures as ours.[3] We additionally compare to published results of these models and several related ones. We evaluate the memory and speed of the ScatResNet-50 model and compare it to the reference models ResNet-50 and the next biggest ResNet model ResNet-101 in the first two rows of Table 2. Our comparisons are done on a single GPU. As in [16,33], we evaluate the inference time of the CNN from the encoding. For memory, we consider memory usage during training as we believe the scattering models are useful for training with fewer resources. We find

[3] http://pytorch.org/docs/0.3.0/torchvision/models.html

that our scattering model has favorable properties in memory and speed usage compared to its non-scattering analogues. In fact, as the next step, we demonstrate large improvements in accuracy, speed, and memory on detection tasks using the ScatResNet-50 network, which indicates that ScatResNet-50 features are also generic for detection.

Table 2: Speed and memory consumption for ImageNet classification sizes (224x224) and detection scale 800px. We compare the inference speed of the learned CNN between the different models and for the detection models the inference speed of feature extraction. To evaluate memory we determine the maximum batch size possible for training on a single GPU. We use a single 11GB Ti 1080 GPU for all comparisons.

	Classification Models		**Detection Models**	
Architecture	Speed (64 images)	Max im. ImageNet	Speed (4 images)	Max im. Coco
Order 1 + ScatResNet-50	0.072	175	0.073	9
ResNet-50	0.095	120	0.104	7
ResNet-101	0.158	70	0.182	2

3.2 Detection experiments

Finally, we apply our hybrid architectures to detection. We base our experiments and hyperparameters on those indicated by the Faster-RCNN implementation of [34] without any specific adaptation to the dataset. We consider both the VOC07 and COCO and adopt the ScatResNet-50 network as the basis of our model. We shared the output of the second layer across a region proposal network and a detection network, which are kept fixed. The receptive field of each output neuron corresponds to 16^2, which is similar to [35,7,36]. The next layers will be fine-tuned for the detection tasks and fed to classification and box-regression layers, as in [35], and a *region proposal network* as done in [36]. Similarly to [7,36], we fixed all the batch normalization [37] layers, including the running means and biases.

Pascal VOC07 Pascal VOC2007 [38] consists of 10k images split equally for training ("train+val") and testing, with about 25k annotations. We chose the same hyperparameters as used in [34]. We used an initial learning rate of 10^{-3} that we dropped by 10 in epoch 5 and we report the accuracy of the epoch 6 in Table 3 on the test set. During training, the images are flipped and rescaled with a ratio between 0.5 and 2, such that the smaller size is 600px as [7,36].
The training procedures used for detection often vary substantially. This includes batch size, weight decay for different parameters, and the number of training

Table 3: Mean average precision on Pascal VOC7 dataset. First-order scattering permits outperforms the related models.

Architecture	mAP
Faster-RCNN Order 1 + ScatResNet-50 (ours)	73.3
Faster-RCNN ResNet-50 (ours)	70.5
Faster-RCNN ResNet-101 (ours)	72.5
Faster-RCNN VGG-16 [34]	70.2

epochs among others. Due to this inconsistency, we train our own baseline models. We use the trained base networks for ResNet-50 and ResNet-101 provided as part of the `torchvision` package in `pytorch` [39] and train the detection models in exactly the same way as described above for ScatResNet50, ResNet-50, and ResNet-101. Table 3 reports a comparison of our ScatResNet model and the ResNet50 and ResNet101 model on this task. The results clearly show that our architecture and base network leads to a substantially better performance in terms of the mAP. On this particular dataset, perhaps due to its smaller size, we find the hybrid model can outperform even models with substantially stronger base networks [17] i.e the performance of ScatResNet-50 is above that of the ResNet101 based model. In the second two rows of Table 2, we show the memory and speed of the different models. The inference speed of the base network feature extractor is shown and for memory, we show the maximum batch size that one can train with. The tradeoff in mAP vs. speed and mAP vs. memory consumption here clearly favors the scattering based models. We now consider a larger scale version of this task on the COCO dataset.

Table 4: Mean average precision on COCO 2015 minival. Our method obtains competitive performance with respect to popular methods.

Architecture	mAP
Faster-RCNN Order 1 + ScatResNet-50	32.2
Faster-RCNN ResNet-50 (ours)	31.0
Faster-RCNN ResNet-101 (ours)	34.5
Faster-RCNN VGG-16 [34]	29.2
Detectron [40]	41.8

COCO We likewise deploy the ScatResNet-50 on the COCO dataset [41]. This detection dataset is more difficult than than PASCAL VOC07. It has 120k images, out of which we use 115k for training and 5k for validation (minival), with 80 different categories. We again follow the implementation of [34] and their

training and evaluation protocol. Specifically, we train the Faster-RCNN networks for 6 epochs with an initial learning rate of 8×10^{-3}, multiplying by a factor 0.1 at epoch 4. We use a minimal size of 800px, and similar scale augmentation w.r.t. Pascal VOC07. We use a batch size of 8 on 4 GPUs and train again all 3 models ScatResNet-50, ResNet-50, and ResNet-101. At test time, we restrict the maximum size to be 1200px as in [34] to permit an evaluation on a single GPU.
Table 4 reports the mAP of our model compared to its non-hybrid counterparts. This score is computed via the standard averaged over IoU thresholds $[0.5, 0.95]$. Our architecture accuracy falls between the one of a ResNet-50 and a ResNet-101. Observing Table 2 the tradeoff in mAP vs. speed and mAP vs. memory consumption here still favors the scattering based models. The results indicate that scattering based models can be favorable even in sophisticated *near*-state-of-the-art models. We encourage future work on combining scattering based models with the *most*-state-of-the-art architectures and pipelines.

4 Conclusion

We consider the problem of compressing an input image while retaining the information and structure necessary to allow a typical CNN to be applied. To the best of our knowledge, this problem has not been directly tackled with an effective solution. We motivate the use of the *first-order scattering* as a candidate for performing the *signal reduction*. We first refine several theoretical results regarding the stability with respect to translation of the first-order scattering. This motivates the use of Gábor wavelets that capture many signal attributes. We then show both on an analytical model and experimentally that reconstruction is possible. We perform experiments on challenging image classification and detection datasets ImageNet and COCO, showing that CNNs approaching the state-of-the-art performance can be built on top of the first-order scattering. This work opens the way to a research on transformations that build compressed input representations. Finally, we incite research on families of wavelets that could increase the resolution reduction and on determining whether our result generalizes to other classes of signals.

Acknowledgements E. Oyallon was supported by a GPU donation from NVIDIA and partially supported by a grant from the DPEI of Inria (AAR 2017POD057) for the collaboration with CWI. S. Zagoruyko was supported by the DGA RAPID project DRAAF. The research presented was also supported by European CHIST-ERA project DELTA, French Ministry of Higher Education and Research, Nord-Pas-de-Calais Regional Council, Inria and Otto-von-Guericke-Universität Magdeburg associated-team north-European project Allocate, and French National Research Agency projects ExTra-Learn (n.ANR-14-CE24-0010-01) and BoB (n.ANR-16-CE23-0003).

References

1. LeCun, Y., Kavukcuoglu, K., Farabet, C., et al.: Convolutional networks and applications in vision. In: International Symposium on Circuits and Systems. (2010) 253–256
2. Williams, T., Li, R.: Wavelet pooling for convolutional neural networks. In: International Conference on Learning Representations. (2018)
3. Rippel, O., Snoek, J., Adams, R.P.: Spectral representations for convolutional neural networks. In: Neural Information Processing Systems. (2015) 2449–2457
4. Le, Q.V.: Building high-level features using large scale unsupervised learning. In: International Conference on Acoustics, Speech and Signal Processing. (2013) 8595–8598
5. Krizhevsky, A., Sutskever, I., Hinton, G.E.: ImageNet classification with deep convolutional neural networks. In: Neural Information Processing Systems. (2012) 1097–1105
6. Simonyan, K., Zisserman, A.: Very deep convolutional networks for large-scale image recognition. `arXiv:1409.1556` (2014)
7. He, K., Zhang, X., Ren, S., Sun, J.: Deep residual learning for image recognition. Computer Vision and Pattern Recognition (2016) 770–778
8. Forsyth, D.A., Ponce, J.: Computer vision: a modern approach. Prentice Hall Professional Technical Reference (2002)
9. Mallat, S.: A wavelet tour of signal processing. Academic Press (1999)
10. Mallat, S., Hwang, W.L.: Singularity detection and processing with wavelets. Transactions on Information Theory **38**(2) (1992) 617–643
11. Skodras, A., Christopoulos, C., Ebrahimi, T.: The jpeg 2000 still image compression standard. Signal Processing Magazine **18**(5) (2001) 36–58
12. Perronnin, F., Larlus, D.: Fisher vectors meet neural networks: A hybrid classification architecture. In: Computer Vision and Pattern Recognition. (2015) 3743–3752
13. Fujieda, S., Takayama, K., Hachisuka, T.: Wavelet convolutional neural networks for texture classification. `arXiv:1707.07394` (2017)
14. Huang, G., Liu, Z., Weinberger, K.Q., van der Maaten, L.: Densely connected convolutional networks. In: Computer Vision and Pattern Recognition. (2017)
15. Levinskis, A.: Convolutional neural network feature reduction using wavelet transform. Elektronika ir Elektrotechnika **19**(3) (2013) 61–64
16. Gueguen, L., Sergeev, A., Liu, R., Yosinski, J.: Faster neural networks straight from JPEG. In: International Conference on Learning Representations Workshop. (2018)
17. Oyallon, E., Belilovsky, E., Zagoruyko, S.: Scaling the scattering transform: Deep hybrid networks. In: International Conference on Computer Vision. (2017)
18. Bruna, J., Mallat, S.: Invariant scattering convolution networks. Transactions on Pattern Analysis and Machine Intelligence **35**(8) (2013) 1872–1886
19. Sánchez, J., Perronnin, F., Mensink, T., Verbeek, J.: Image classification with the fisher vector: Theory and practice. International Journal of Computer Vision **105**(3) (2013) 222–245
20. Morel, J.M., Yu, G.: ASIFT: A new framework for fully affine invariant image comparison. Journal on Imaging Sciences **2**(2) (2009) 438–469
21. Olshausen, B.A., Field, D.J.: Emergence of simple-cell receptive field properties by learning a sparse code for natural images. Nature **381**(6583) (1996) 607
22. Mallat, S.: Group invariant scattering. Communications on Pure and Applied Mathematics **65**(10) (2012) 1331–1398

23. Mallat, S., Waldspurger, I.: Phase retrieval for the cauchy wavelet transform. Journal of Fourier Analysis and Applications **21**(6) (2015) 1251–1309
24. Waldspurger, I., d'Aspremont, A., Mallat, S.: Phase recovery, maxcut and complex semidefinite programming. Mathematical Programming **149**(1-2) (2015) 47–81
25. Krajsek, K., Mester, R.: A unified theory for steerable and quadrature filters. In: Advances in Computer Graphics and Computer Vision. (2007) 201–214
26. Soulard, R.: Ondelettes analytiques et monogènes pour la représentation des images couleur. PhD thesis, Université de Poitiers (2012)
27. Delprat, N., Escudié, B., Guillemain, P., Kronland-Martinet, R., Tchamitchian, P., Torresani, B.: Asymptotic wavelet and gabor analysis: Extraction of instantaneous frequencies. Transactions on Information Theory **38**(2) (1992) 644–664
28. Bruna, J., Mallat, S.: Audio texture synthesis with scattering moments. `arXiv:1311.0407` (2013)
29. Bruna, J.: Scattering representations for recognition. PhD thesis, École Polytechnique (2013)
30. Lindeberg, T.: Feature detection with automatic scale selection. International Journal of Computer Vision **30**(2) (1998) 79–116
31. Lowe, D.G.: Distinctive image features from scale-invariant keypoints. International Journal of Computer Vision **60**(2) (2004) 91–110
32. Zagoruyko, S., Komodakis, N.: Wide residual networks. In: British Machine Vision Conference. (2016)
33. Torfason, R., Mentzer, F., Agustsson, E., Tschannen, M., Timofte, R., Van Gool, L.: Towards image understanding from deep compression without decoding. `arXiv:1803.06131` (2018)
34. Yang, J., Lu, J., Batra, D., Parikh, D.: A faster `pytorch` implementation of faster R-CNN. https://github.com/jwyang/faster-rcnn.pytorch (2017)
35. Girshick, R.B.: Fast R-CNN. International Conference on Computer Vision (2015) 1440–1448
36. Ren, S., He, K., Girshick, R., Sun, J.: Faster R-CNN: towards real-time object detection with region proposal networks. Transactions on Pattern Analysis and Machine Intelligence **39**(6) (2017) 1137–1149
37. Ioffe, S., Szegedy, C.: Batch normalization: Accelerating deep network training by reducing internal covariate shift. `arXiv:1502.03167` (2015)
38. Everingham, M., Van Gool, L., Williams, C.K., Winn, J., Zisserman, A.: The pascal visual object classes (VOC) challenge. International journal of computer vision **88**(2) (2010) 303–338
39. Paszke, A., Gross, S., Chintala, S., Chanan, G., Yang, E., DeVito, Z., Lin, Z., Desmaison, A., Antiga, L., Lerer, A.: Automatic differentiation in pytorch. (2017)
40. He, K., Gkioxari, G., Dollár, P., Girshick, R.: Mask R-CNN. In: International Conference Computer Vision. (2017) 2980–2988
41. Lin, T.Y., Maire, M., Belongie, S., Hays, J., Perona, P., Ramanan, D., Dollár, P., Zitnick, C.L.: Microsoft COCO: Common objects in context. In: European Conference on Computer Vision. (2014) 740–755